Laborator si seminar

Programare in Java si software matematic

Adrian Ichimescu

Adrianichimescu@gmail.com

Math in Java

mXpareser is a super easy, rich, fast and highly flexible math expression parser library:
- Parser and evaluator of mathematical expressions
- Formulas provided as plain text/string

Software delivers easy to use API for JAVA.

mXparser
How to install?

Where to download it from?
Is available on different sources:

- mxParser web site
- maven

To install it define a new Java project using IDE Eclipse wizard and import it as a new Library->Add External JARs

Or if the Project is already created – right click on the project -> Build Path -> Libraries ->Add External JARs

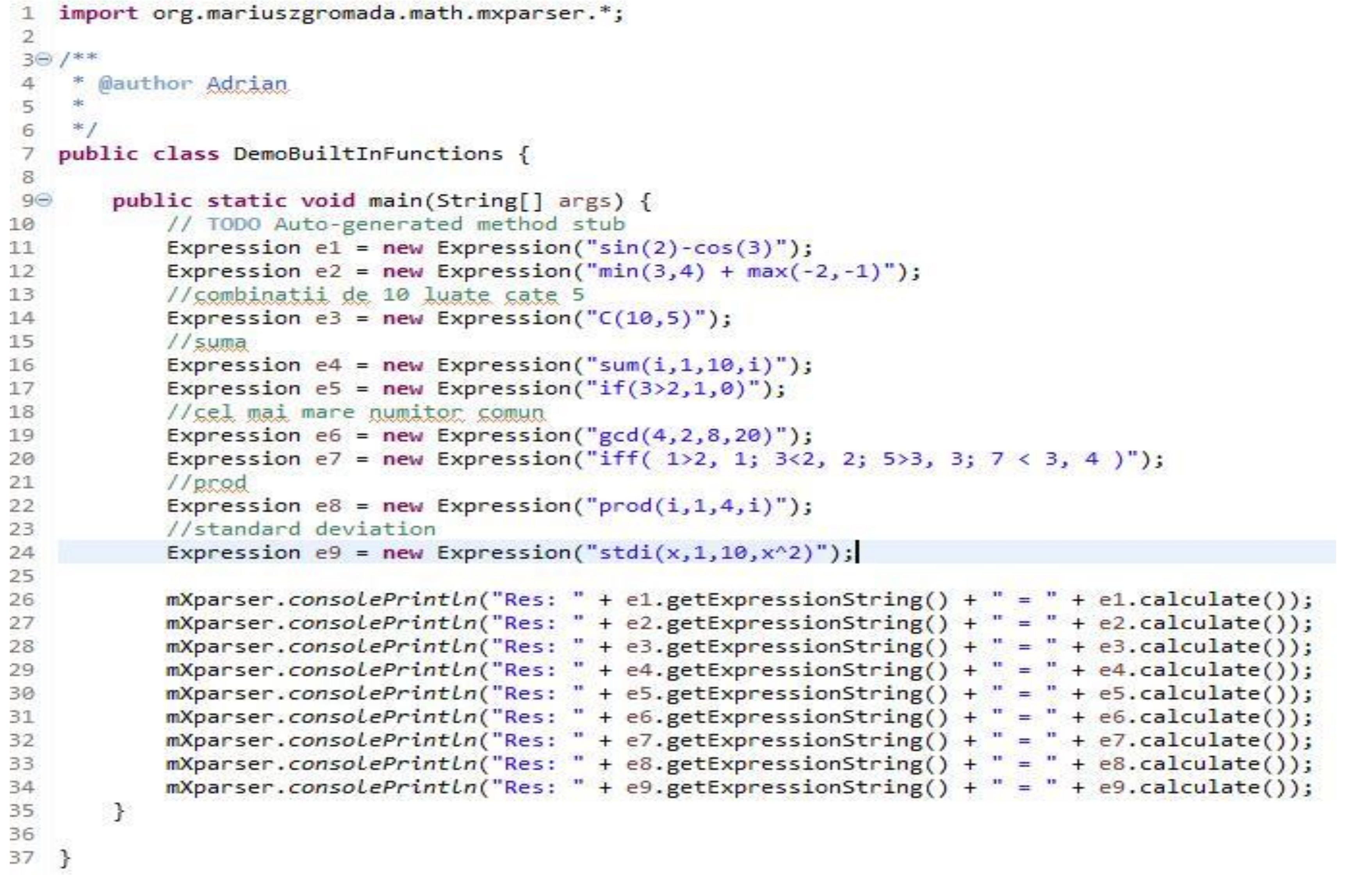
Here is a demo in YouTube

What is mxParser offers to us?

- Rich built-in library of operators, constants, math functions, differentiation and integration
- User defined: arguments, functions, recursive functions and general recursion
- Grammar and internal syntax checking
- Random numbers, variable and probability distributions

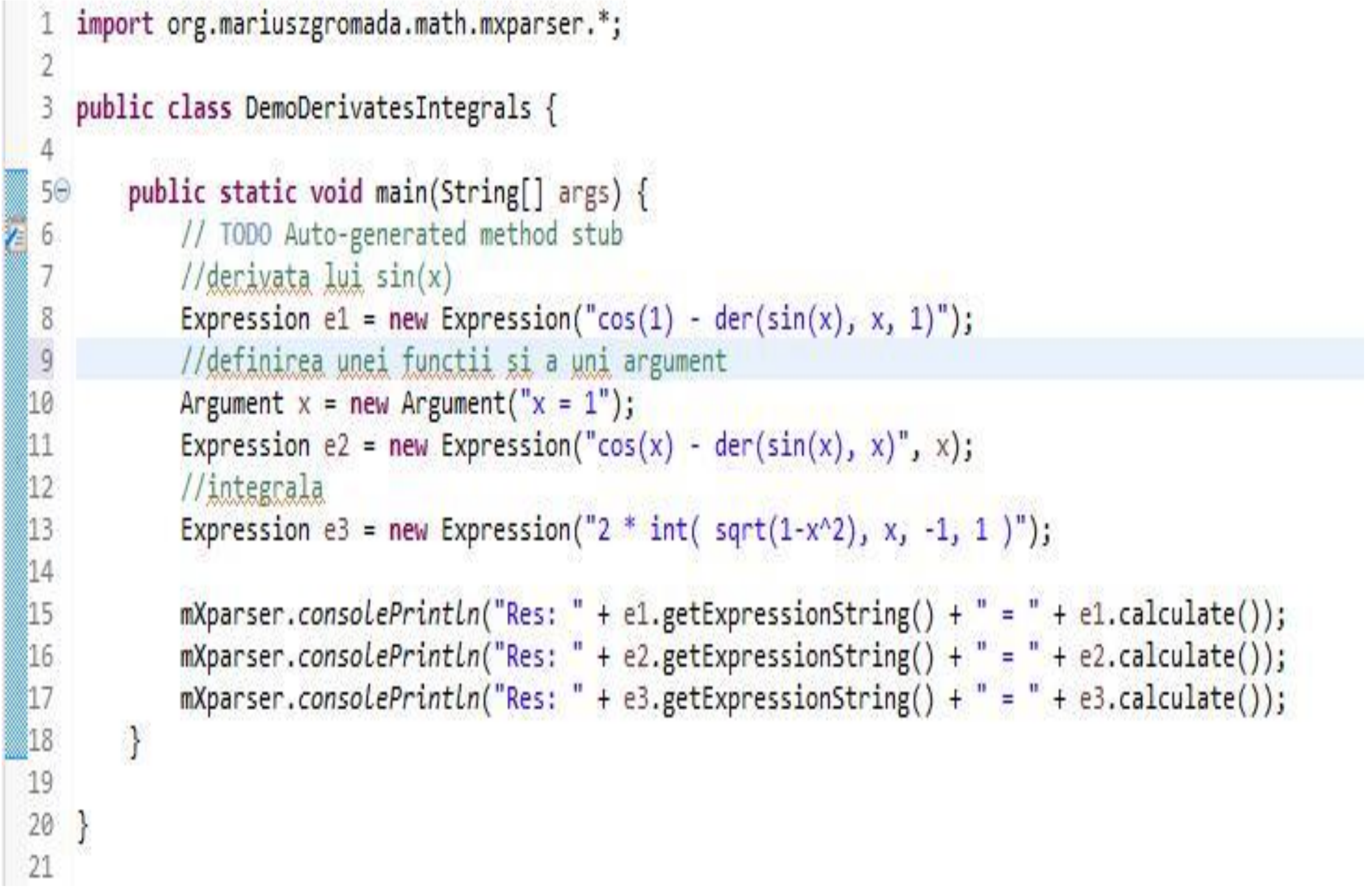
Full details here available

mXparser examples BuiltInFunctions



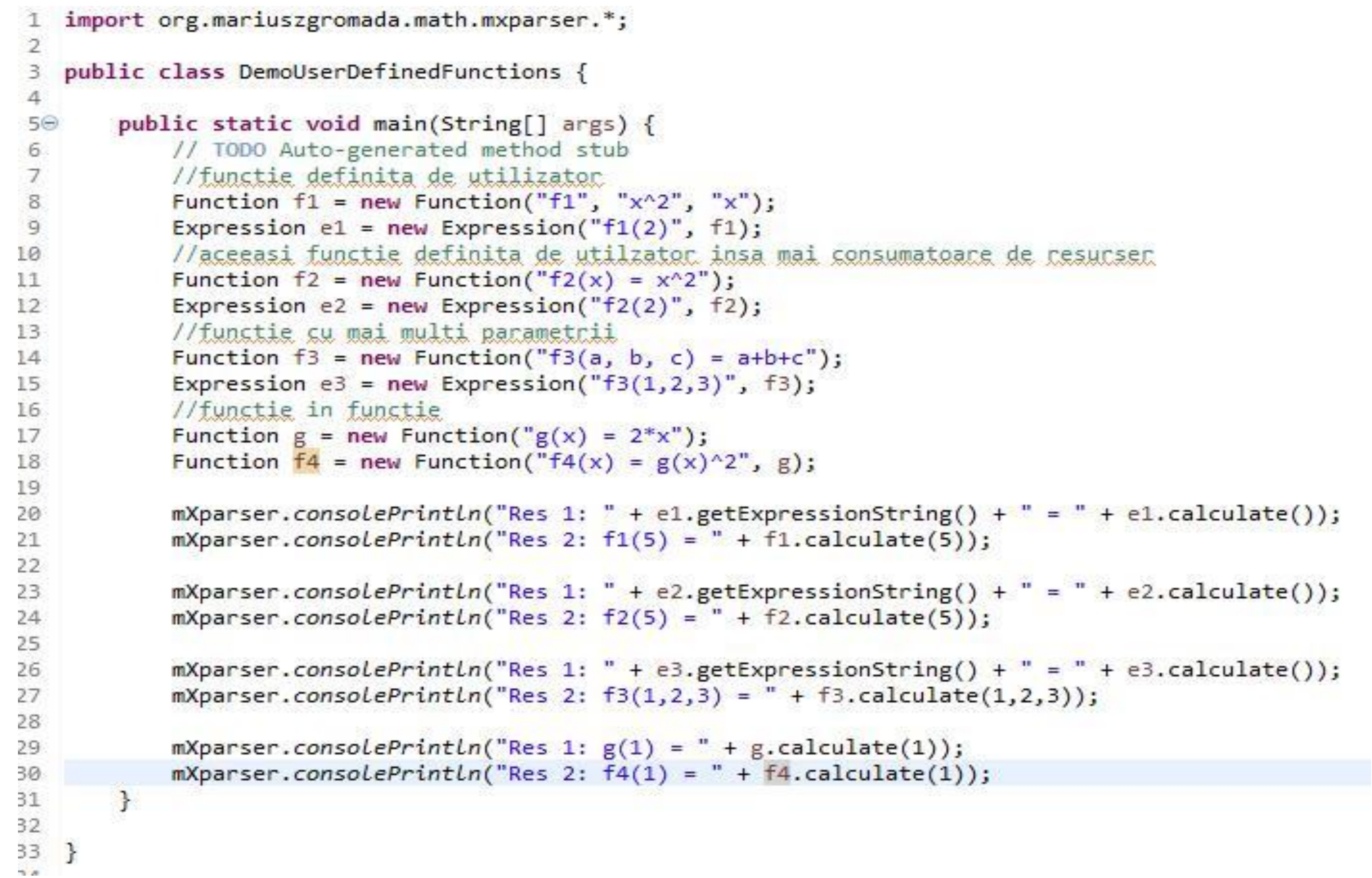
```java
import org.mariuszgromada.math.mxparser.*;

/**
 * @author Adrian
 *
 */
public class DemoBuiltInFunctions {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Expression e1 = new Expression("sin(2)-cos(3)");
        Expression e2 = new Expression("min(3,4) + max(-2,-1)");
        //combinatii de 10 luate cate 5
        Expression e3 = new Expression("C(10,5)");
        //suma
        Expression e4 = new Expression("sum(i,1,10,i)");
        Expression e5 = new Expression("if(3>2,1,0)");
        //cel mai mare numitor comun
        Expression e6 = new Expression("gcd(4,2,8,20)");
        Expression e7 = new Expression("iff( 1>2, 1; 3<2, 2; 5>3, 3; 7 < 3, 4 )");
        //prod
        Expression e8 = new Expression("prod(i,1,4,i)");
        //standard deviation
        Expression e9 = new Expression("stdi(x,1,10,x^2)");

        mXparser.consolePrintln("Res: " + e1.getExpressionString() + " = " + e1.calculate());
        mXparser.consolePrintln("Res: " + e2.getExpressionString() + " = " + e2.calculate());
        mXparser.consolePrintln("Res: " + e3.getExpressionString() + " = " + e3.calculate());
        mXparser.consolePrintln("Res: " + e4.getExpressionString() + " = " + e4.calculate());
        mXparser.consolePrintln("Res: " + e5.getExpressionString() + " = " + e5.calculate());
        mXparser.consolePrintln("Res: " + e6.getExpressionString() + " = " + e6.calculate());
        mXparser.consolePrintln("Res: " + e7.getExpressionString() + " = " + e7.calculate());
        mXparser.consolePrintln("Res: " + e8.getExpressionString() + " = " + e8.calculate());
        mXparser.consolePrintln("Res: " + e9.getExpressionString() + " = " + e9.calculate());
    }

}
```

mXparser Derivates Integrals



```java
import org.mariuszgromada.math.mxparser.*;

public class DemoDerivatesIntegrals {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //derivata lui sin(x)
        Expression e1 = new Expression("cos(1) - der(sin(x), x, 1)");
        //definirea unei functii si a uni argument
        Argument x = new Argument("x = 1");
        Expression e2 = new Expression("cos(x) - der(sin(x), x)", x);
        //integrala
        Expression e3 = new Expression("2 * int( sqrt(1-x^2), x, -1, 1 )");

        mXparser.consolePrintln("Res: " + e1.getExpressionString() + " = " + e1.calculate());
        mXparser.consolePrintln("Res: " + e2.getExpressionString() + " = " + e2.calculate());
        mXparser.consolePrintln("Res: " + e3.getExpressionString() + " = " + e3.calculate());
    }

}
```

mXparser User Defined Functions



```java
import org.mariuszgromada.math.mxparser.*;

public class DemoUserDefinedFunctions {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        //functie definita de utilizator
        Function f1 = new Function("f1", "x^2", "x");
        Expression e1 = new Expression("f1(2)", f1);
        //aceeasi functie definita de utilzator insa mai consumatoare de resurser
        Function f2 = new Function("f2(x) = x^2");
        Expression e2 = new Expression("f2(2)", f2);
        //functie cu mai multi parametrii
        Function f3 = new Function("f3(a, b, c) = a+b+c");
        Expression e3 = new Expression("f3(1,2,3)", f3);
        //functie in functie
        Function g = new Function("g(x) = 2*x");
        Function f4 = new Function("f4(x) = g(x)^2", g);

        mXparser.consolePrintln("Res 1: " + e1.getExpressionString() + " = " + e1.calculate());
        mXparser.consolePrintln("Res 2: f1(5) = " + f1.calculate(5));

        mXparser.consolePrintln("Res 1: " + e2.getExpressionString() + " = " + e2.calculate());
        mXparser.consolePrintln("Res 2: f2(5) = " + f2.calculate(5));

        mXparser.consolePrintln("Res 1: " + e3.getExpressionString() + " = " + e3.calculate());
        mXparser.consolePrintln("Res 2: f3(1,2,3) = " + f3.calculate(1,2,3));

        mXparser.consolePrintln("Res 1: g(1) = " + g.calculate(1));
        mXparser.consolePrintln("Res 2: f4(1) = " + f4.calculate(1));
    }

}
```

mXparser
all available

https://github.com/mariuszgromada/MathParser.org-mXparser

Homework

- Define function f(x,y)=sqrt(sin(x)^2+cos(y)^2) si calculate expresia f(1,2)-10
- Calculate the average value and standard deviation of the following array 1, 3, 5, 7, 9, 11, 13, 15, 17, 18
- Calculate the least common multiple of the following array 3, 9, 12, 2, 8, 6, 18